

Rancang Bangun Modul PAD (*Packet Assembler Dissassembler*) Menggunakan AX.25 pada Sistem Komunikasi ITS-SAT

Pasang Arung Padang, Eko Setijadi, dan Gamantyo Hendrantoro
Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS)
Jl. Arief Rahman Hakim, Surabaya 60111
e-mail: ekoset@ee.its.ac.id

Abstrak— ITS-Sat merupakan jenis satelit piko yang saat ini sedang dikembangkan di Institut Teknologi Sepuluh Nopember. ITS-Sat membutuhkan suatu protokol komunikasi yang mengatur tata cara komunikasi satelit. Protokol AX.25 merupakan protokol komunikasi yang digunakan dalam sistem komunikasi ITS-Sat. Protokol ini berada pada *layer data link* dalam model OSI *Layer*. Protokol ini dapat membangun dan memutuskan *link* serta melakukan pengiriman data. Protokol AX.25 menggabungkan data-data *field* menjadi suatu *frame*. Makalah ini mengimplementasikan protokol AX.25 ke dalam modul PAD (*Packet Assembler Dissassembler*). Protokol AX.25 diimplementasikan ke dalam mikrokontroler yang merupakan otak dari modul PAD dengan menggunakan bahasa pemrograman. Modul PAD berfungsi untuk mengkapsulasi data tiap *field* menjadi sebuah *frame* AX.25 sebelum dikirim dan kemudian *frame* AX.25 yang diterima dienkapsulasi kembali menjadi data *field*. Modul PAD yang dibuat dapat mengirim dan menerima data teks sebanyak 500 karakter. Hasil pengujian menunjukkan bahwa modul PAD yang dibuat mampu melakukan komunikasi antar modul PAD dengan baik.

Kata Kunci— ITS-Sat, Modul PAD, Protokol AX.25, data-link

I. PENDAHULUAN

Teknologi satelit adalah salah satu teknologi yang perkembangannya sangat pesat. Institut Teknologi Sepuluh Nopember merupakan salah satu perguruan tinggi yang merencanakan untuk mengembangkan ITS-Sat secara mandiri. ITS-Sat yang akan dikembangkan dapat mengirim citra maupun data telemetri. Data dalam hal ini harus memiliki bentuk khusus untuk dapat ditransmisikan. Salah satu metodenya adalah mengubah data ke dalam bentuk paket radio. Salah satu protokol yang mengatur paket radio adalah AX.25. Protokol AX.25 yang merupakan protokol komunikasi berada pada *layer 2* dalam model OSI *Layer* sehingga protokol ini dapat membangun dan memutuskan *link* serta dapat melakukan pengiriman data. Protokol AX.25 dapat melakukan komunikasi *half-duplex* maupun *full-duplex* dengan baik. Pada protokol AX.25 setiap *frame* mengandung alamat penerima dan pengirim, sehingga mampu bekerja dengan baik untuk koneksi langsung antar perangkat, dan juga memungkinkan untuk melakukan koneksi lebih dari satu perangkat[1].

Protokol AX.25 ini diimplementasikan ke dalam sebuah mikrokontroler dengan menggunakan bahasa pemrograman bahasa C. Mikrokontroler merupakan otak dari modul PAD (*Packet Assembler Dissassembler*) untuk mengirim dan menerima data. Data yang diterima berupa data

teks yang diinputkan oleh *keyboard* dan pada penerima dan ditampilkan pada LCD. Data yang dilewatkan oleh protokol AX.25 berupa *frame*.

Raharjo dalam tugas akhirnya [2] telah merancang modul PAD AX.25 untuk sistem komunikasi nano dengan menggunakan sistem mikrokontroler 1280 yang terdiri dari 2 modul yaitu modul rangkaian mikrokontroler Atmega 128 dan modul rangkaian sistem minimum mikrokontroler Atmega128. Sistem dengan desain 2 modul ini tidak sesuai untuk aplikasi pada satelit piko karena ukuran *relatif* besar. Oleh sebab itu tugas akhir ini bertujuan merancang modul PAD dengan ukuran *relatif* lebih kecil.

Makalah ini melaporkan mengenai desain dan implementasi protokol AX.25 ke dalam modul PAD untuk sistem komunikasi ITS-Sat. Bab II menjelaskan mengenai model dan struktur dari protokol AX.25 sebagai landasan untuk perancangan modul PAD. Tahapan perancangan dan implementasi termasuk didalamnya perancangan perangkat keras dan perancangan perangkat lunak dari modul PAD dipaparkan pada Bab III, sedangkan hasil pengujian dan pengukuran perangkat serta kesimpulan dilaporkan pada Bab IV dan Bab V.

II. PROTOKOL AX.25

A. Model AX.25

Amateur X.25 atau yang dikenal dengan protokol AX.25 adalah protokol *layer data link* dalam OSI *Layer Reference*. Sebagai protokol yang bekerja pada *layer data link* protokol AX.25 bertanggungjawab untuk membangun dan memutuskan *link*, serta melakukan *transfer* informasi. Protokol AX.25 dapat dimodelkan sebagai berikut[1] :

Layer	Function(s)	
	Segmenter	Management
Data Link (2)	Data Link	Data Link
	Link Multiplexer	
Physical (1)	Physical	
	Silicon/Radio	

Gambar. 1. Model AX.25

Protokol AX.25 berbeda dengan protokol *layer 2* pada umumnya karena pada protokol AX.25 tidak terdapat perbedaan kelas antar perangkat yang terhubung tidak seperti

kol *layer 2* lainnya. Pada protokol AX.25 kedua ujung *link* dianggap setara dan dapat memulai pengiriman *frame* sendiri.

B. Struktur Frame AX.25

Protokol AX.25 mengirimkan data dalam bentuk *frame*. Dan tiap *frame* tersusun dari bagian-bagian yang lebih kecil (*field*). Setiap *field* mempunyai fungsi khusus dan terdiri dari jumlah *byte*. Dalam protokol AX.25 terdapat tiga tipe *frame* yang dikenal, yaitu *frame* informasi (*frame I*), *frame* pengawas atau *supervisory frame* (*frame S*) dan *frame* tidak bernomor atau *Unnumbered frame* (*frame U*).

Flag	Address	Control	Info	FCS	Flag
8 Bits	112/224 Bits	8/16 Bits	N*8 Bits	16 Bits	8 Bits

Gambar. 2. Konstruksi *Frame U* dan *S*[1]

Flag	Address	Control	PID	Info	FCS	Flag
8 Bits	112/224 Bits	8/16 Bits	8 Bits	N*8 Bits	16 Bits	8 Bits

Gambar. 3. Konstruksi *Frame Information*[1]

Struktur *frame* AX.25 terdiri dari :

- **Field Flag**

Field flag berfungsi untuk menentukan awal dan akhir dari suatu *frame*. *Field* ini memiliki pola bit yang unik yaitu 01111110 atau 7E hexa.

- **Field Address**

Field address digunakan untuk menyatakan alamat penerima dan pengirim serta *reapeter*. *Field* ini terdiri dari 7 oktet, yang terdiri dari 6 oktet *callsign* yang merupakan susunan dari karakter alfanumerik ASCII (*American Standard Code for Information Interchange*), dan 1 oktet berisi nomor pengenalan stasiun sekunder atau *Secondary Station Identifier* (SSID).

- **Control Field**

Field ini berfungsi untuk menentukan tipe *frame* yang digunakan.

- **PID Field**

Field PID (*Protocol Identifier*) ini mengidentifikasi model layer 3 yang digunakan dalam *frame*.

- **Info Field**

Field ini digunakan untuk menyampaikan informasi dari satu sisi hubungan ke sisi lainnya. *Field* ini berisi data yang akan dikirimkan.

- **FCS Field**

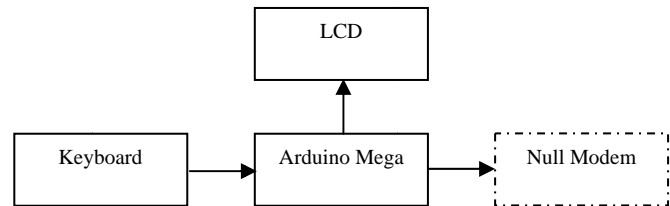
FCS (*Frame Check Sequence*) merupakan bilangan 16 bit yang dikalkulasikan oleh pengirim dan penerima suatu *frame* berdasarkan *field* alamat, kontrol, PID, dan Informasi. FCS berfungsi untuk mengetahui kerusakan yang terjadi pada *frame* saat melalui pengiriman.

III. PERANCANGAN DAN IMPLEMENTASI

Perancangan modul PAD dengan menggunakan protokol AX.25 terdiri 2 tahap, yaitu : perancangan perangkat keras dan perancangan perangkat lunak.

A. Perancangan Perangkat Keras

Perangkat keras satelit piko ITS-Sat didesain dengan dimensi 10cm x10cm x 10cm dan berat 1 kg. Perangkat keras yang dibutuhkan untuk perancangan ini adalah arduino mega, *keyboard*, dan LCD. Namun selain itu pada perancangan ini juga menggunakan perangkat tambahan seperti laptop, kabel, dan konektor.

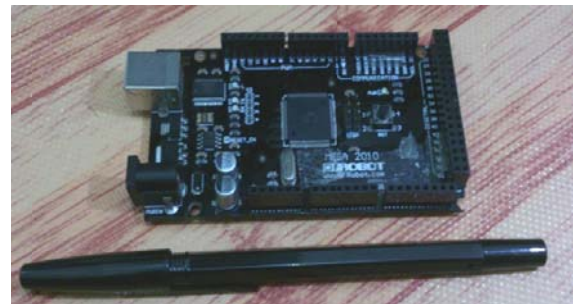


Gambar. 4. Blok diagram sistem PAD

Perangkat keras yang digunakan dalam perancangan ini adalah sebagai berikut :

- **Arduino mega**

Arduino mega mempunyai kapasitas memori 128kB, dimana 4kB digunakan sebagai bootloader[3]. Arduino mega mempunyai ukuran *board* dengan panjang 9,7 cm dan lebar 5,3 cm, serta mempunyai berat 35 gram. Sehingga cukup untuk diterapkan pada sistem komunikasi satelit piko ITS-Sat.



Gambar. 5. Arduino mega

- **LCD**

Dalam pemilihan LCD dalam perancangan ini memperhatikan ukuran dari LCD untuk menyesuaikan ukuran dari *board* arduino. LCD yang digunakan dalam perancangan ini mempunyai ukuran yang lebih kecil yaitu 2 x 16 karakter namun dapat menampilkan data yang dilewatkan.



Gambar. 6. LCD 2 x16 karakter

- **Keyboard**

Keyboard yang digunakan dalam perancangan ini berfungsi sebagai perangkat yang memasukkan data pada perancangan modul PAD.

Pada perancangan ini *keyboard* dan LCD digunakan sebagai alat bantu dalam proses pengujian untuk memasukkan

dan menampilkan data. Perangkat-perangkat yang disebutkan sebelumnya kemudian dirangkai. Dan dengan menggunakan alat bantu laptop program protokol AX.25 yang dirancang kemudian diimplementasikan ke dalam mikrokontroler.

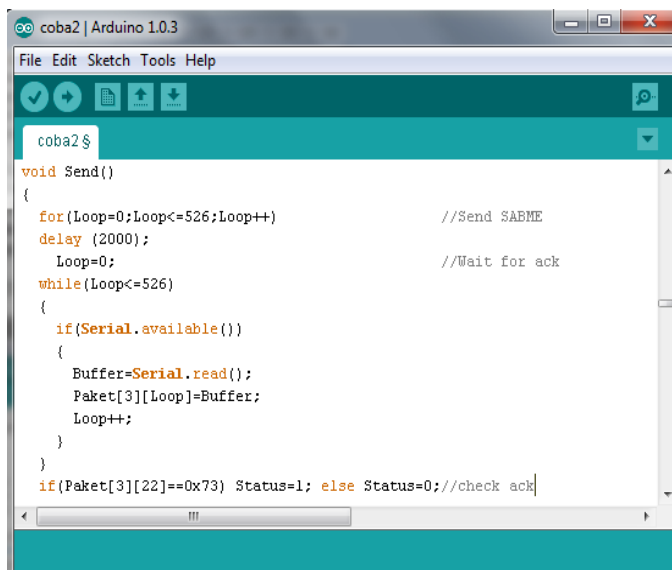
B. Perancangan Perangkat Lunak

Pada perancangan ini menggunakan *software* dari arduino untuk pembuatan program dan mengimplementasikan ke dalam mikrokontroler. Pada perancangan perangkat lunak ini mampu membaca data teks dari *keyboard* dan ditampilkan pada LCD, serta mampu untuk mengkapsulasi data setiap *field* menjadi suatu *frame* AX.25. Perangkat lunak ini juga dirancang untuk dapat mengenkapsulasi *frame* AX.25 yang diterima dan kemudian ditampilkan pada LCD.

Perancangan ini meliputi perancangan program proses memulai hubungan komunikasi dan pengiriman informasi.

a. Memulai hubungan komunikasi

Proses ini terdiri dari *frame* SABME (*Set Asynchronous Balance Mode Extended*) yang berfungsi untuk meminta hubungan komunikasi. Dan *frame* UA (*Unnumbered Acknowledge*) berfungsi memberikan *acknowledge* atas penerimaan *frame* SABME.



Gambar. 7. Tampilan program memulai hubungan

Untuk memulai komunikasi *frame* SABME dikirim dan *timer* pada mikrokontroler mulai menghitung untuk menunggu jawaban berupa *frame* UA. Apabila sampai *timer* berhenti menghitung belum menerima *frame* UA, maka pengirim akan mengirim ulang *frame* SABME.

Frame SABME diatur dengan parameter sebagai berikut :

- *Field address* dimasukkan secara manual melalui *keyboard* berisi alamat dan SSID dari stasiun tujuan, stasiun pengirim, dan stasiun *repeater* yang dimasukkan melalui *keyboard*.
- *Field Control* bernilai hexa 7F berfungsi untuk mengirimkan perintah SABME.
- *Field PID* diberi hexa F0 karena hanya menggunakan *layer* 2.

- *Field Info* diset kosong atau diisi nilai spasi (20 hexa).
- *Field FCS* bernilai hexa FFFF.

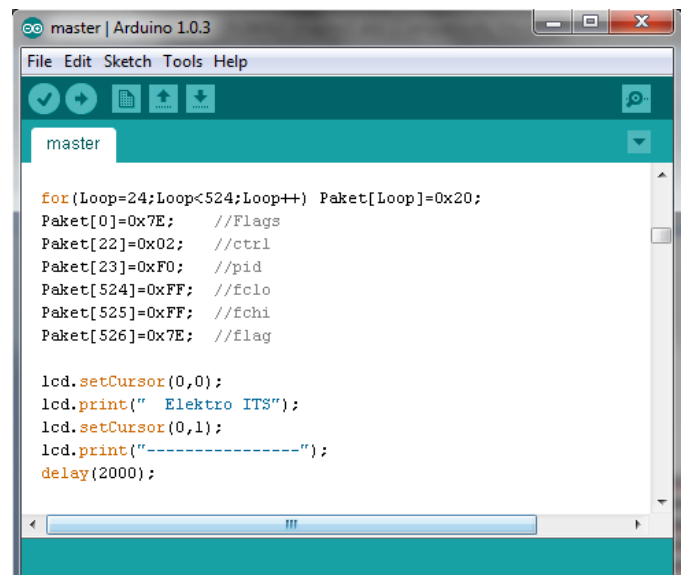
Untuk *frame* UA diatur dengan parameter sebagai berikut :

- *Field address* berisi alamat amatir stasiun tujuan, alamat amatir dari stasiun pengirim dan alamat amatir dari *repeater* yang dimasukkan melalui *keyboard*.
- *Field Control* bernilai hexa 73 untuk mengirimkan perintah UA.
- *Field PID* bernilai hexa F0 karena hanya menggunakan *layer* 2.
- *Field Info* diset kosong atau diisi nilai spasi (20 hexa).
- *Field FCS* diset hexa FFFF.

b. Pengiriman informasi

Pada perancangan ini perangkat lunak dirancang untuk dapat mengirim dan menerima *frame* yang berisi informasi. *Frame* informasi diatur dengan parameter sebagai berikut :

- *Field address* dimasukkan secara manual melalui *keyboard* berisi alamat dan SSID dari stasiun tujuan, stasiun pengirim, dan *repeater*.
- *Field Control* bernilai hexa 02 untuk menginformasikan bahwa data yang dikirim merupakan data dengan urutan 1.
- *Field PID* bernilai hexa F0 karena hanya menggunakan *layer* 2.
- *Field Info* diisi dengan informasi yang dikirimkan.
- *Field FCS* diset hexa FFFF.



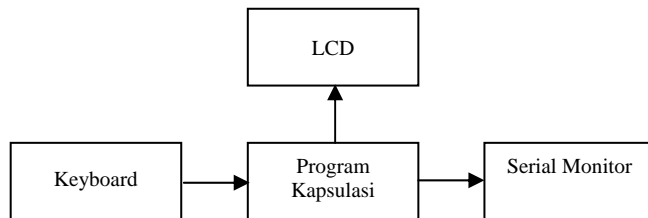
Gambar. 8. Tampilan program tiap *field* pada *frame* informasi

IV. PENGUJIAN DAN ANALISA

A. Pengujian Kapsulasi Data

Pengujian *kapsulasi* data dilakukan untuk mengetahui kemampuan fungsi *assembler* dari modul PAD. Pada bagian ini data masukan melalui *keyboard* dan ditampilkan pada

LCD, namun sebelumnya data diolah terlebih dahulu oleh mikrokontroler dengan program *kapsulasi*, dan kemudian ditampilkan pada *serial monitor* untuk melihat hasil *kapsulasi* dari masing-masing *field*.

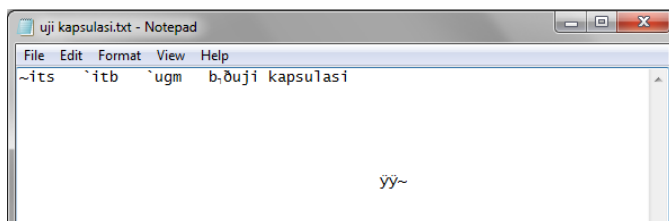


Gambar. 9. Blok diagram pengujian kapsulasi

Pada pengujian ini data dari setiap *field* pada *frame* yang akan dikapsulasi diatur sebagai berikut :

- Alamat *receiver* diisi “its”
- SSID dari *receiver* diisi “0”
- Alamat *transmitter* diisi “itb”
- SSID dari *transmitter* diisi “0”
- Alamat *repeater* diisi “ugm”
- SSID dari *repeater* diisi “1”
- *Control* diberi nilai 02 hexa
- *PID* diberi nilai F0 hexa
- *Field* informasi (*message*) diisi dengan kalimat “uji kapsulasi”.
- *FCS* diberi nilai hexa FFFF.

Data dari tiap *field* yang telah dimasukkan kemudian digabung menjadi satu *frame*. *Frame* hasil kapsulasi dapat dilihat pada *serial monitor*. *Frame* yang ada pada *serial monitor* kemudian disimpan dalam bentuk file dengan ekstensi .txt. Sehingga dapat mudah dalam pembacaan. Gambar 10 menunjukan bahwa data masing-masing *field* berhasil dikapsulasi menjadi sebuah *frame* AX.25.



Gambar. 10. Hasil kapsulasi *frame*

B. Pengujian Handshaking

Pada bagian ini kedua modul PAD tetap dihubungkan dengan kabel, namun pada pengujian ini pin TX dan RX yang ada pada masing-masing modul digunakan. Karena akan terjadi kirim mengirim *frame*. Modul pengirim akan mengirimkan *frame* SABME (*Set Asynchronous Balance Mode*) untuk meminta hubungan komunikasi. Penerima akan menerima *frame* SABME dan kemudian akan mengirimkan *frame* UA (*Unnumbered Acknowledge*) sebagai jawaban atas penerimaan *frame* SABME.

Frame SABME yang dikirim oleh modul penerima diatur dengan ketentuan sebagai berikut :

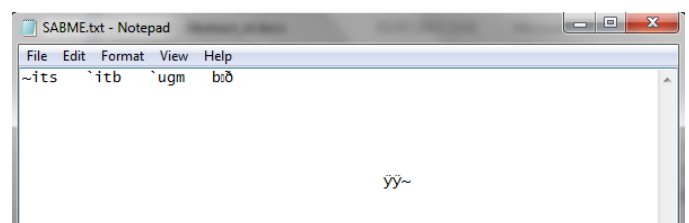
- Alamat *receiver* diisi “its”
- SSID dari *receiver* diisi “0”
- Alamat *transmitter* diisi “itb”

- SSID dari *transmitter* diisi “0”
- Alamat *repeater* diisi “ugm”
- SSID dari *repeater* diisi “1”
- *Control* bernilai hexa 7F
- *PID* diset F0.
- *Field* Informasi diset kosong atau diisi spasi.
- *Field FCS* diset kosong atau diisi nilai hexa FFFF.

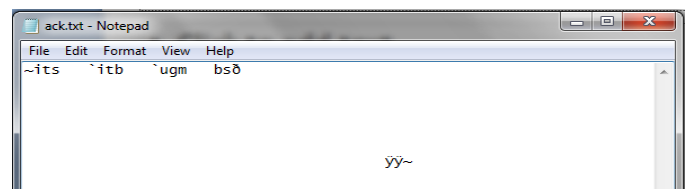
Sedangkan *frame* UA diatur dengan parameter sebagai berikut:

- Alamat *receiver* diisi “its”
- SSID dari *receiver* diisi “0”
- Alamat *transmitter* diisi “itb”
- SSID dari *transmitter* diisi “0”
- Alamat *repeater* diisi “ugm”
- SSID dari *repeater* diisi “1”
- *Control* bernilai hexa 73
- *PID* diset F0.
- *Field* Informasi diset kosong atau diisi spasi.
- *Field FCS* diset kosong atau diisi nilai hexa FFFF.

Berdasarkan hasil *frame* yang ditampilkan oleh *serial monitor* seperti yang ditunjukkan pada gambar 11 dan 12, dapat disimpulkan bahwa proses *handshaking* antar modul telah berhasil.



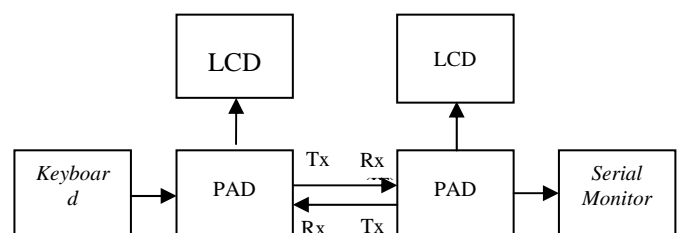
Gambar. 11. Hasil *frame* SABME



Gambar. 12. Hasil *Frame* UA

C. Pengujian Fungsi Assembler Disassembler

Pada pengujian ini modul pengirim berfungsi mengkapsulasi data masing-masing *field* menjadi *frame* sedangkan modul penerima berfungsi memecah *frame* menjadi *field* dan ditampilkan pada LCD.

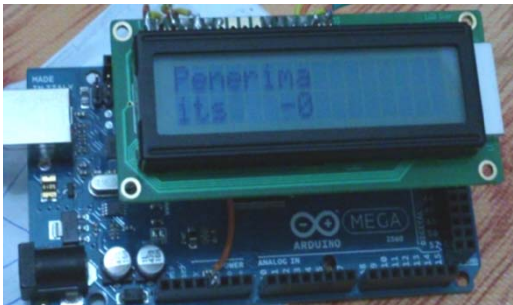


Gambar. 13. Blok diagram pengujian

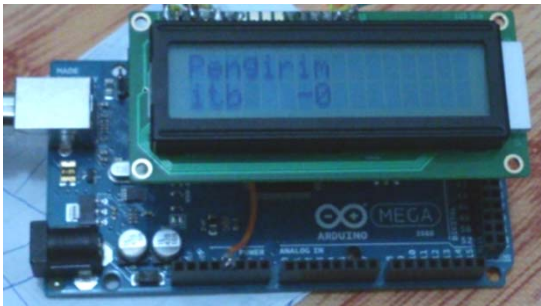
Data *field* yang ditampilkan pada LCD di penerima merupakan data masukan *keyboard* pada modul pengirim. Pada pengujian ini struktur *frame* diatur sebagai berikut :

- Alamat *receiver* diisi “its”
- SSID dari *receiver* diisi “0”
- Alamat *transmitter* diisi “itb”
- SSID dari *transmitter* diisi “0”
- Alamat *repeater* diisi “ugm”
- SSID dari *repeater* diisi “1”
- *Control* diberi nilai 02 hexa
- *PID* diberi nilai F0 hexa
- *Field* informasi (*message*) diisi dengan kalimat “uji koneksi”.
- *FCS* diberi nilai FFFF hexa.

Gambar 14 merupakan tampilan alamat *receiver* “its” dengan SSID “0” yang diterima oleh modul penerima dan ditampilkan pada LCD. Alamat *transmitter* “itb” dengan SSID “0” yang diterima kemudian ditampilkan pada LCD, seperti pada gambar 15.

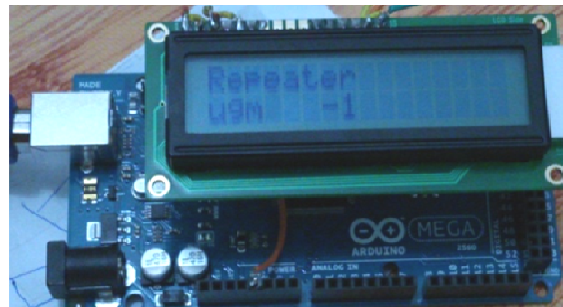


Gambar. 14. Tampilan alamat *receiver* dan SSID pada penerima



Gambar. 15. Tampilan alamat *transmitter* dan SSID yang diterima

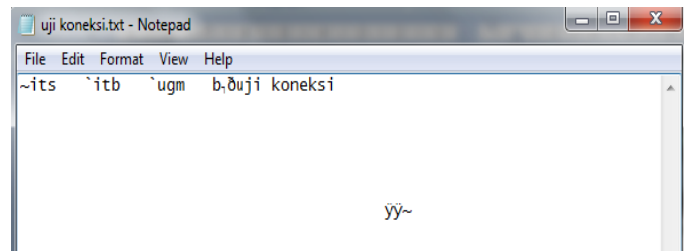
Pada gambar 16 menunjukkan alamat *repeater* “ugm” dengan SSID “1” yang ditampilkan pada LCD penerima. *Field* informasi yang diterima ditampilkan berurutan per 16 karakter tiap baris karena modul PAD menggunakan LCD dengan ukuran 16x2 seperti pada gambar 17. Setelah baris kedua ditampilkan secara otomatis baris kedua naik ke baris pertama, sedangkan baris pertama sebelumnya dihapus. Hal ini akan berlangsung terus menerus hingga semua isi dari *field* informasi ditampilkan. Hasil *frame* yang diterima secara penuh ditampilkan oleh *serial monitor* seperti yang ditunjukkan pada gambar 18.



Gambar. 16. Tampilan alamat *repeater* beserta SSID pada penerima



Gambar. 17. Tampilan *field* informasi pada penerima

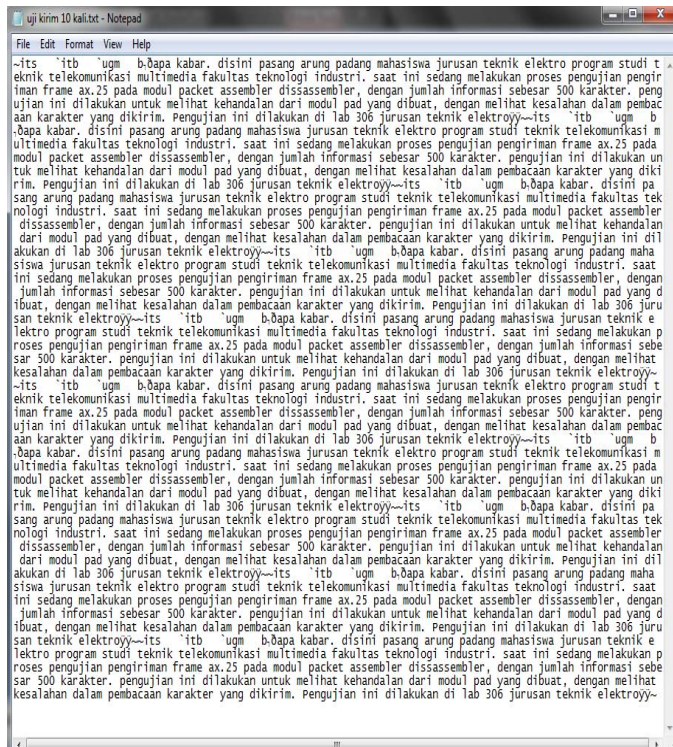


Gambar. 18. Tampilan struktur *frame* pada *serial monitor*

Pada pengujian ini dilakukan sebanyak 10 kali pengiriman, dengan jumlah *field* informasi sebanyak 500 karakter. *Field* informasi dibentuk dengan beberapa kalimat sehingga terjadi variasi karakter. Pengiriman karakter yang berbeda-beda bertujuan untuk melihat karakter salah dalam pembacaan.

Pada pengujian ini struktur *frame* yang dikirim diatur dengan parameter yang sama dengan pengujian fungsi *Assembler-Dissassembler*, hanya saja *field* informasi diisi dengan kalimat “apa kabar. disini pasang arung padang mahasiswa jurusan teknik elektro program studi teknik telekomunikasi multimedia fakultas teknologi industri. saat ini sedang melakukan proses pengujian pengiriman frame ax.25 pada modul packet assembler dissassembler, dengan jumlah informasi sebesar 500 karakter. pengujian ini dilakukan untuk melihat kehandalan dari modul pad yang dibuat, dengan melihat kesalahan dalam pembacaan karakter yang dikirim. Pengujian ini dilakukan di lab 306 jurusan teknik elektro”.

Gambar 19 menunjukkan hasil pengujian yang dilakukan sebanyak 10 kali dengan menggunakan *baud rate* 9600 yang diatur pada modul pengirim dan penerima. Dari gambar 19 dapat dilihat tidak ada terjadi kesalahan dalam pembacaan karakter dalam proses pengiriman. Hal ini berbanding terbalik apabila *baud rate* yang diatur pada modul pengirim dan penerima berbeda. Semua karakter yang dikirim mengalami kesalahan pembacaan saat diterima oleh modul penerima.



Gambar. 19. Hasil pengiriman sebanyak 10 kali

D. Diskusi

Perancangan dan implementasi protokol AX.25 ke dalam modul PAD untuk sistem komunikasi ITS-Sat dilakukan dengan menggunakan mikrokontroler berupa modul Arduino Mega. Perancangan modul PAD dimulai dengan membuat blok diagram dan *flow chart*. Blok diagram digunakan untuk merangkai modul arduino mega dengan *peripheral* tambahan sehingga membentuk suatu sistem PAD yang dapat memasukkan dan menampilkan data, sedangkan *flowchart* yang digambarkan dibuat dalam bentuk program dengan *software* Arduino yang menggunakan bahasa C. Program yang selesai dibuat kemudian diimplementasikan ke dalam modul PAD.

Pengujian modul PAD terdiri dari pengujian kapsulasi data, pengujian *handshaking*, dan pengujian fungsi *assembler-dissassembler*. Modul PAD yang dibuat telah teruji dapat membentuk suatu *frame* serta melakukan pengiriman informasi antar modul PAD. Modul PAD dapat mengirim *frame* dengan jumlah informasi sebanyak 500 karakter tanpa kesalahan saat kedua modul diatur dengan *baud rate* 9600 bps. Pengujian yang dilakukan menunjukkan bahwa modul PAD mampu melakukan pengiriman data serta memiliki ukuran yang *relatif* kecil sehingga memenuhi kriteria untuk diimplementasikan pada satelit piko ITS-Sat.

V. KESIMPULAN

Modul PAD menggunakan protokol AX.25 yang telah dibuat dapat digunakan pada sistem komunikasi ITS-Sat untuk mengkapsulasi dan enkapsulasi data. Modul PAD dapat mengirimkan informasi sebanyak 500 karakter dengan baik tanpa adanya kesalahan penerimaan data pada penerima.

Dengan ukuran yang *relatif* kecil serta kemampuan pengiriman data, modul PAD dapat diimplementasikan pada sistem komunikasi satelit piko ITS-Sat.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada tim penelitian strategis nasional 2012 Kemdikbud “Pengembangan stasiun bumi untuk komunikasi data, citra dan video dengan satelit LEO VHF/UHF/S-band menuju kemandirian teknologi satelit” yang telah memberikan dukungan finansial.

DAFTAR PUSTAKA

- [1] Beech, William A., Nielsen, Douglas E., Taylor, Jack., “AX.25 Link Acces Protocol v2.2”, TAPR. 1997.
- [2] Raharjo, Dicky Rismawan, “Rancang Bangun Modul PAD (*Packet Assembler Disassembler*) Untuk Stasiun Bumi Portabel Dengan Menggunakan Protokol Ax.25 Pada Komunikasi Satelit Inusat-1” Institut Teknologi Sepuluh Nopember Surabaya, 2012.
- [3] “Arduino mega” : <http://arduino.cc/en/Main/arduinoBoardMega>